Technical notes on the phylobase package

Ben Bolker & other phylobase authors

August 20, 2009

1 Introduction

This vignette is intended as a depository for technical notes of primary interest to phylobase developers and others interested in gory details/what's going on under the hood — by contrast with the other phylobase vignette, which is more of an introduction & user's manual.

2 Notes on other formats and "round-trip" issues

We should in principle be able to convert from other formats to `phylo4`(`d`) and (`ape::phylo`, `ouch::ouch` etc.) and back without losing any information. The two classes of exceptions would be (1) where `phylo4` stores *less* information than the other formats (we would try to avoid this), and (2) where there are ambiguities etc. in the other formats (we would try to avoid this, too, but it may be difficult; ideally we would consult the package maintainers and try to get them to eliminate the ambiguities in their formats).

Ideally we would be able to use `identical()` to test equality — this tests "bit-by-bit" equality, and is intolerant of *any* differences in format. More loosely, `all.equal()` allows for numeric variation below a certain tolerance, etc. (these correspond to `RUnit::checkEquals()` and `RUnit::checkIdentical()`).

Case in point: `ape` is not entirely consistent in its internal representations, which causes some difficulty in creating perfect round trips (see `tests/roundtrip.R` for workarounds). In particular,

- `unroot()` contains several statements that subtract 1 from components of the data structure that were previously stored as `integer`. Because 1 is subtracted an not `1L` (an explicitly integer constant), this coerces those elements to be `numeric` instead.
- different ways of creating trees in `ape` (`read.tree()`, `rcoal()`) generate structures with the internal elements in different orders. When `phylobase` re-exports them, it always uses the order {`edge`, `edge.length`, `tip.label`, `Nnode`, [`node.label`], [`root.edge`]}, which matches the trees produced by `rcoal` but not those produced by `read.tree`

- because of differences in ordering standards, it's not clear that we can always preserve ordering information through non-trivial manipulations in phylobase

3 phylobase tools, coding standards, etc.

We use `camelCase`; try to indent our code according to the rules in the R extensions manual; and use `<-` for assignment.

We are in the process of moving our testing infrastructure to the `RUnit` framework.

Our development infrastructure hosted by r-forge, http://r-forge.r-project.org/projects/phylobase/:

- SVN repository
- mailing list (`phylobase-devl@lists.r-forge.r-project.org`)
- bug & feature tracking
- to install the development version:

  ```
  > install.packages("phylobase", repos = "http://r-forge.r-project.org")
  ```

Production version of the package (will soon be) hosted at CRAN: you can install it via

```
> install.packages("phylobase")
```